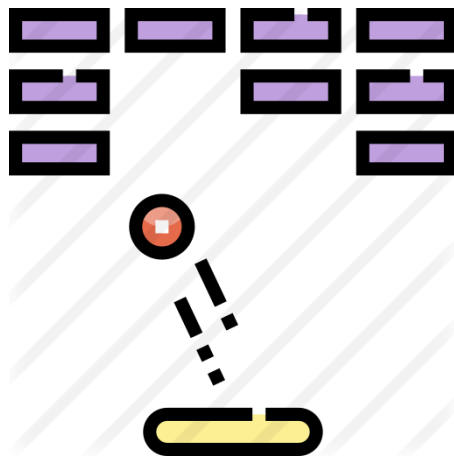


Manual Técnico

Arkanoid



Realizado por:

Mario Andres Calix Sosa, 00309119
Juan Humberto Gallardo Hernandez, 00091813
Kevin Alexander Cea Patriz, 00203919

Contenido

Manual Técnico Arkanoid.....	1
Aspectos Generales.....	3
Objetivo del documento.....	3
Descripción General.....	3
Software Utilizado.....	3
Modelos Utilizados.....	4
Diagrama de clases.....	4
Diagrama Entidad-Relación.....	6
Diagrama Relacional.....	6
Conceptos técnicos.....	7
Nomenclaturas.....	8
Eventos y Excepciones.....	9
Excepciones.....	9
Eventos.....	10

Aspectos generales

Objetivo del documento:

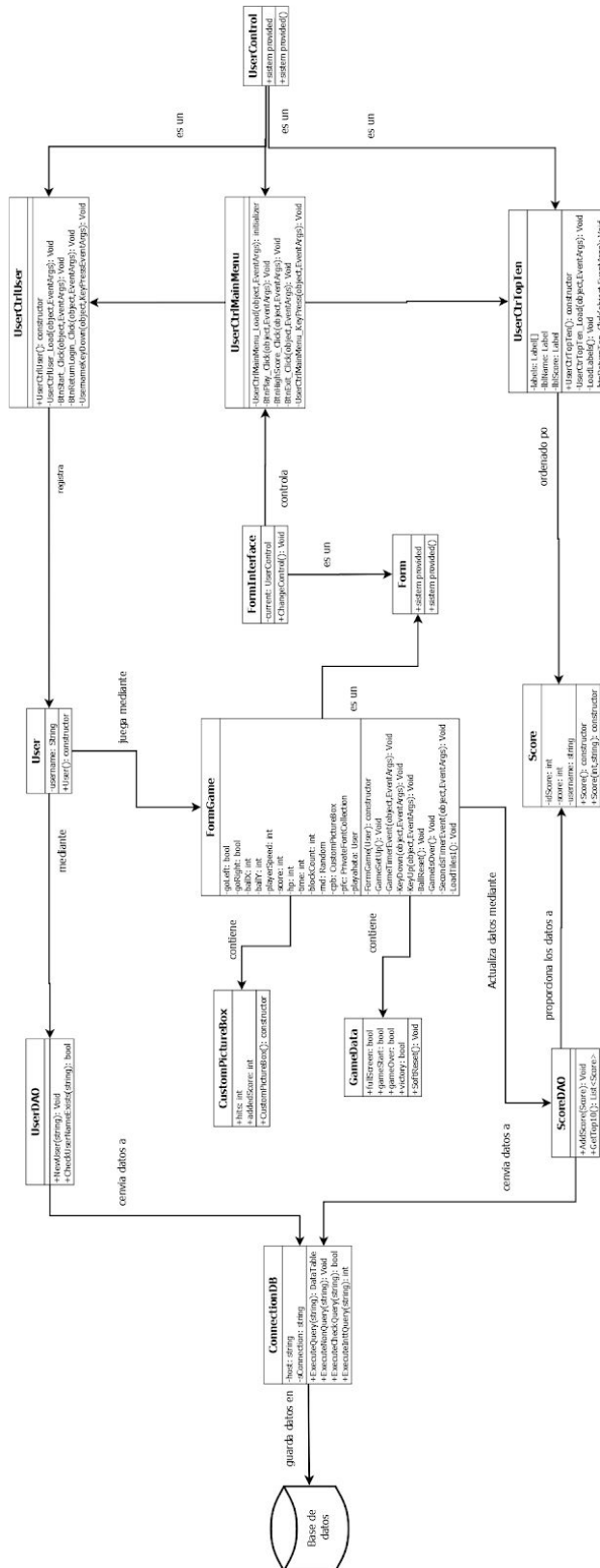
En el presente documento se explica el proceso de recreación del videojuego Arkanoid, dando a conocer herramientas y metodología usadas.

Descripción general:

El programa consiste en la recreación del famoso juego de arcade de la década de los 80's, Arkanoid. Distintos usuarios pueden disfrutar de un solo nivel de este, los mejores se muestran en el top 10. Para su creación se utilizó el patrón de arquitectura <<modelo -vista - controlador>>

Software utilizado:

Para la creación del juego se utilizó el lenguaje de programación C# y JetBrains Rider 2019 como IDE. Como base de datos se utilizó PostgreSQL 12 con Npgsql 4.1 para su respectiva conexión con el lenguaje. Se utilizó también la herramienta de photoshop en línea para diversos aspectos estéticos.

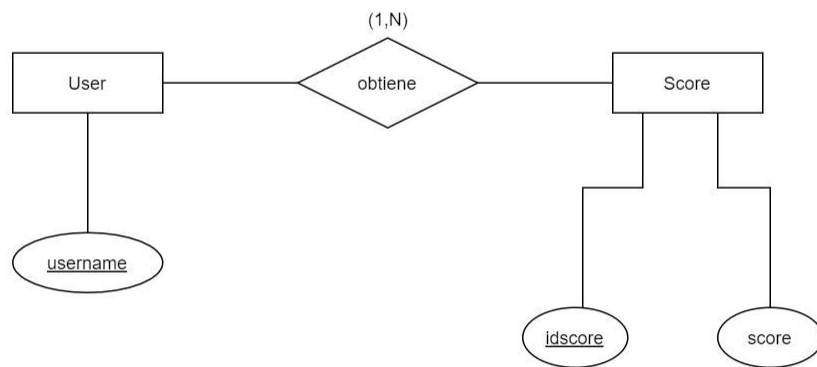


Explicando el diagrama de clases: la aplicación carga en primer lugar el `FormInterface` como ventana principal y padre de 3 controles de usuario `UserCtrlMainMenu`, `UserCtrlTopTen` y `UserCtrlUser`. Como su nombre lo indica `MainMenu` se encargará de desplegar las opciones del menú principal por medio de 3 botones, cada uno con su evento único. Con estos botones se cargará el `UserCtrlUser`, el `UserCtrlTopTen` o terminará la aplicación.

Si se detecta click en `btnHighScore` se cambiará el user control a `TopTen` y se desplegarán (si existen) labels con las puntuaciones más altas. Si se detecta click en `btnPlay` se cargará el user control `User`, en el cual se le pedirá por medio de un `textBox` el nombre de usuario que empezará el juego. Si todo está conforme y se detecta click o presiona la tecla `Intro/Enter` entonces por medio de `UserDAO` se guardará el nombre del usuario y comenzará el juego.

El juego se encuentra en el formulario `FormGame` el cual hace uso de la mayoría de clases de controlador para los datos del juego (posición del jugador y pelota, bloques, puntuación, relojes del juego, etc.). El juego termina si se cumplen dos condiciones, ya sea el jugador termina sin vidas, se desplegará un mensaje de juego fallido y se almacenará la puntuación que tenga; o todos los bloques son destruidos. En tal caso se desplegará un mensaje de juego completado y se aplicará un multiplicador a su puntuación para luego almacenarla en la BD por medio de `ScoreDAO`. El usuario tiene entonces dos opciones, reintentar el juego o salir al menú principal por medio de dos eventos de `KeyDown` (salir con `Esc`; reintentar con `Enter`).

2. Diagrama entidad relación extendido:



3. Diagrama relacional normalizado:



Conceptos técnicos

La interfaz gráfica consiste en 2 formularios/ventanas globales FormInterface y FormGame. FormInterface siendo el formulario padre de 3 controles de usuario (user control): UserCtrlMainMenu, UserCtrlTopTen y UserCtrlUser. La aplicación empieza entonces cargando el formulario FormInterface y a partir de ahí se cargan los UC y posteriormente el formulario FormGame.

Clases vista (Interfaz gráfica)

UserCtrlTopTen.cs
UserCtrlUser.cs
UserCtrlMainMenu.cs
FormInterface.cs
FormGame.cs

Clases modelo

User.cs
Score.cs
GameData.cs
CustomPictureBox.cs

Clases Controlador

UserDAO.cs
ScoreDAO.cs
ConnectionDB.cs

Nomenclaturas

Para identificar de una manera fácil los componentes del entorno gráfico se ha utilizado la siguiente normativa de nombramiento:

<Abreviatura_de_tipo>Descripcion<id_correlativo>

Componente	Abreviación
Label	lbl
TextBox	txt
Button	btn
Exception	ex
User Control	userCtrl
Custom PictureBox	cpb
Random	rnd

Eventos y Excepciones

Excepciones:

Las excepciones constan de un constructor que recibe un mensaje (string) de error y son las siguientes:

1. `EmptyUsernameException.cs`: Esta excepción se da cuando el usuario intenta ingresar al juego dejando el campo de username vacío.
2. `LengthExceededException.cs`: Esta excepción ocurre cuando el usuario intenta ingresar al juego con un nombre que excede los caracteres permitidos por la aplicación.
3. `MissingFontException.cs`: Esta excepción ocurre cuando se intenta correr el programa pero no se tiene instalada la fuente personalizada.
4. `NoScoreSavedException.cs`: Esta excepción se da cuando no hay puntajes guardados en la aplicación y se ingresa a la pantalla de TopTen (nadie ha jugado el juego).
5. `NullUserControlException.cs`: Esta excepción ocurre cuando se intenta cargar un user control que no existe o es nulo (nunca debería de ocurrir en el programa).

Eventos:

Los eventos dentro de la aplicación son autoexplicativos debido a que todos tratan con clicks y eventos de teclas. Los controles del juego están regidos por eventos de tecla tanto keyDOWN como keyUP. La funcionalidad de los botones está dada con clicks y eventualmente keyDOWN de la tecla Intro/Enter.